

APPLICATIONS MANAGER WITH VARIABLE MANAGEMENT
INSTRUCTION SET

INSA1

The invention relates to an information processing system making it possible to process data originating from at least one application, comprising an applications manager executing a management instruction set. The invention also relates to a digital decoder receiving in particular applications by way of a bouquet of television programs.

2. Description of Prior Art
An information processing system can be a machine which makes it possible to process data originating from an application. The application can be a collection of data. The data generally constitute a string of instructions formulated in a programming language. The information processing system can be constructed by using in particular an operating system and an execution system receiving data of an application. The information processing system can also comprise other systems making it possible to manage peripherals attached thereto and generally anything which is not taken onboard by the operating and execution systems.

SUMMARY OF THE INVENTION
An information processing system according to the invention makes it possible to process data originating from at least one application and comprises:

- an operating system for executing the application,
- an execution system, and
- an applications manager which can execute at least one variable management instruction set so as to influence the operating system and/or the execution system in particular when the application is executed or when switching from the execution of the application to another execution of another application.

A first embodiment of the invention makes provision for the information processing system to comprise a means for loading the variable management

A second embodiment of the invention makes provision for the source of management instructions to be the application itself.

A digital decoder according to the invention receives at least one application by way of data relating to services from a digital stream and comprises:

- A fourth embodiment of the invention makes provision for the variable management instruction set to be of the static declarative kind. The management instruction set describes functions relating to a state or to a transition from an executing application to another program. Each application can contain in a preamble a management instruction set which is of the static declarative kind.

35 A sixth embodiment of the invention makes provision for the applications manager to comprise a means of selecting the variable management instruction set which selects a management instruction set in accordance with at least one criterion determined so

that the selected management instruction set has priority of execution.

BRIEF DESCRIPTION OF THE DRAWINGS

In what follows, exemplary embodiments of the invention are described so as to afford a better understanding thereof. Reference is made to Figures 1 to 3:

- Figure 1 containing a simplified diagram of an information processing system;

- Figure 2 containing a simplified diagram of a digital decoder;

- Figure 3 containing a simplified diagram of another digital decoder.

DETAILED DESCRIPTION

An information processing system represented in Figure 1 can be constructed by using an operating system 1. The operating system 1 comprises software making it possible to manage tasks, to allocate space in a memory and to address peripheral devices in conjunction with the information processing system.

An execution system 2 receives, directly or indirectly by way of a memory, data of an application 3. The execution system 2 makes it possible to execute the string of instructions conveyed by the data. The execution system 2 can be constructed with the aid of software. The execution system 2 communicates with the operating system 1 so as to access in particular the peripheral devices and a memory (not represented) of the information processing system.

The information processing system can comprise an applications manager 4. The latter makes it possible to execute a management instruction set. Thus, the applications manager 4 makes it possible to influence the operating system 1 and/or the execution system 2 when the application is executed for example. It would for example be possible for the applications manager 4 to indicate to the operating system 1 what priorities to give to commands originating from the execution system 2 when the application is executed.

The operating system, the execution system and the applications manager are, according to the present

A digital decoder 5 for television 6 represented in Figure 2 makes it possible to receive an application 3 by way of a satellite receiver 7, a cable network 8 and/or a hertzian antenna 9. The decoder is for example a decoder meeting the DVB and MPEG II standards. The application is transmitted in a multiplexed digital stream, the latter not necessarily transporting an audiovisual television program. It is also possible to receive other applications on other multiplexes. Moreover, it is also possible to receive applications by way of a digital channel modulated on an analog signal and time-division multiplexed with an analog television signal, but in what follows we shall generally be concerned with the case of a totally digital system.

30 The digital decoder 5, and more especially, the assembly formed by the operating system 1 and the virtual machine 10, can be designed to execute several applications in a multitask manner, that is to say at the same time.

35 The digital decoder 5 furthermore comprises hardware and/or software components (not represented) such as one or more drivers so that the operating system can communicate with peripheral devices, a user interface allowing a user to communicate with the

application 3 executed or with the digital decoder 5 and optionally comprising one or more function keys, a memory making it possible to store the application 3, possible other applications or graphical data, etc. The
5 decoder can also comprise decoding means (MPEG II audio and video decoding according to the present example) making it possible to decode a demultiplexed stream of audiovisual data from a multiplexed digital stream and to transmit the decoded video to the television 6.

10 The application manager 4 makes it possible to execute a management instruction set and communicates with the virtual machine 10 and the operating system 1. It carries out functions which are not taken on board either by the virtual machine 10, or by the operating
15 system 1.

The functions resulting from the execution of the management instruction set are for example the following:

- consideration of a state of the executing
20 application when a change of transponder (corresponding to a multiplexed stream) or of service occurs. The change of transponder/service can be brought about for example by a user, by the application itself or even by a broadcaster (which are not represented) which
25 broadcasts the content of the streams. The applications manager 4 can, for example, interrupt the executing application or place it on standby. The applications manager 4 can freeze the last picture displayed on the television or display a specified graphic while the
30 change of transponder and/or of the service is effected. This may be necessary to fill in time while loading another application from a stream from the new transponder or associated with another service;
- starting a specified procedure when a change
35 of application has not been performed within a specified time span;
- configuring function keys and rendering them active or otherwise;

09007044 03004

- 5 The management instruction set is stored in a management memory (not represented in Figure 2) and cannot be modified during normal use of the digital decoder 5. The management instruction set is relatively voluminous and complex. Its formulation deploys a considerable development effort. Thus, each time a modification of the management instruction set is required to obtain a different manner of operation of the applications manager 4, it is necessary for a manufacturer or for a programmer of the applications manager 4 to re-embark on a new development of a complete set of management instructions and on a new configuration of the digital decoder 5, in particular the replacement or the total reprogramming of the applications manager 4, this possibly entailing major costs.

It would also be advantageous to be able to update the application manager 4 while avoiding having to install a new configuration in the digital decoder through intervention by the manufacturer on the digital decoder.

The application manager 4 comprises a variable management instruction set 11, that is to say one which can be modified, exchanged or erased at any moment.

The variable management instruction set 11 is executed by the application manager 4, this resulting

in a number of functions which are implemented via communication with the operating system 1 and the virtual machine 10. These functions can be the same as those described earlier in this description. However, the list of functions described is not exhaustive. It is simply intended to explain through examples the role of the applications manager 4.

The variable management instruction set 11 can be stored in a rewritable memory, for example, in a random access memory. A loading means 12 makes it possible to load the variable management instruction set 11 to the applications manager 4. The loading means 12 can be linked to one or more sources of management instructions; for example a user interface 13 of the digital decoder 5, a direct link 14 with a source of the applications, an application link 15 with the application 3 itself. In the latter case, the variable management instruction set 11 can be contained in a preamble 16 of the application 3. The preamble 16 is a first part of the application 3 received by the digital decoder 5. Having received the variable management instruction set 11, the application manager 4 can execute these instructions and carry out corresponding functions while the application 3 is being loaded in full.

Moreover, the decoder can comprise a default instruction set, which is short-circuited by an instruction set loaded later, if certain criteria, for example priority criteria, are fulfilled. This short-circuiting can be associated with one or more applications. In this case, the default instruction set is not erased, but remains available for other applications.

The loading means 12 is for example a digital packet demultiplexer of the MPEG II Systems type received by way of the direct link 14. The source of applications may be multifold: a server linked to the decoder 5 via the switched telephone network, a satellite, cable or hertzian digital or analog

broadcasting network, etc. The necessary circuits for reception and demodulation are not illustrated, since they are well known per se to those skilled in the art. The existence of a preamble 16 does not necessarily
5 entail the existence of the application 3. It is conceivable to include a management instruction set in the preamble 16 and to transmit the latter to the loading means 12, even without there being an associated application.

10 In the case where the source of the management instructions is the direct link 14 with a source of the applications, it is possible for a broadcaster of the application to supply a specific set of management instructions for his applications. The latter set may
15 for example entail the application manager 4 displaying a graphic characteristic of the broadcaster during the waiting time caused by the loading of an application.

In the case where the source of the management instructions is the user interface 13, it is possible
20 for a user to determine for example the functions underlying certain tasks of the digital decoder 5. As already mentioned, if an instruction set local to the decoder exists, it may be short-circuited under condition by a loaded set.

25 In the case where no external source such as the user interface 13, the direct link 14 or the application link 15 supplies management instructions, provision may be made to use a standard management instruction set stored permanently in the applications
30 manager 4.

In an advantageous embodiment there is provision for the management instruction set originating from different sources to be given
35 priorities for execution, according to a predetermined criterion. Thus, it may for example be defined that a management instruction set originating via the application link 15 has priority over an instruction set originating via the direct link 14 with a source of the applications. The applications manager receiving or

0000704 052004
F02090 F02090

having received management instruction sets from these two links 14 and 15, gives priority to the execution of that originating from the application link 15.

The variable instruction set 11 can have a variable volume. For example, provision could be made for the latter to comprise management instructions originating from several sources of management instructions. Thus, if the decoder allows the execution of several applications in parallel, it is possible for the application manager 4 to carry out different functions for each executing application.

An example of the behavior of a decoder will be described in what follows.

According to this example, the applications manager comprises the following instructions:

- Display a boot-up bitmap
- Set the video screen to black
- Freeze the video image
- Define the keys managed by the application at the outset (group of keys of the remote control)
- Take the focus if possible
- Enable audio/video
- Disable audio/video

The parameters supplied in respect of or with a given application are:

- Boot-up bitmap (optional)
- Group of keys
- Priority of the application

It is assumed that initially the state of the decoder is the following:

- Audio/video in progress: yes
- Priority with the foreground application ("possessing the focus"): 1 (navigator)
- Applications executing:

Name	Supplier	Priority	Focus
Weather	Broadcaster X	2	No
Navigator	Decoder manufacturer	1	Yes

In the case of the present example, the navigator is an application built into the decoder at the outset and allowing the user to implement the decoder.

5 A request for focus on the part of an
application signifies according to the present example
that this application is requesting to be executed in
the foreground. The other applications may nevertheless
be executed in parallel, in the "background", if the
10 system is multitask.

A new application is then loaded, for example a telepurchasing application, also supplied by the broadcaster X, this loading being triggered by the detection of the broadcasting of the application in the digital stream received by the decoder.

- New application:

Name	Provider	Priority	Focus request
Shop	Broadcaster X	2	Yes

```

    The (default) static instruction set of the
20 decoder is:

```

```

    Define the keys managed by the application on
start up (group of keys)

```

If Request Focus Then Take the focus if possible

25 (Remark: the possibility of taking the focus
depends on the priority of the application which made
the request relative to that possessing the focus)

30 The instruction set present in the signal and positioned by the broadcaster X for the Shop application is the following:

Set the video plane to black

```

        If  audio/video  in  progress  Then  disable
audio/video

```

```

        Define the keys managed by the application on
35  start up ({Quit, P+, P-})

```

```

        If Request Focus Then Take the focus if
possible

```

Instruction set present in the application:

Enable audio/video

The following dynamic behavior results from this collection of sets:

5 1. The application is being initiated, an instruction set must be applied (before initiation of the application). The broadcaster has given an instruction set for this application which has priority over the default instruction set of the terminal. It is
10 therefore the set of the application which is applied.

2. The video plane is set to black.

3. The audio/video which is currently playing is stopped.

15 4. The Quit, P+ and P- keys of the remote control will not be managed by the terminal when the application has the focus.

5. Focus is requested, but denied since the Shop application has lower priority than the application having the focus (Navigator)

20 6. The application is initiated (without the focus)

7. The application applies its complementary instruction set and enables a new audio/video stream.

The new state of the decoder is then:

25 • Audio/video in progress: yes
 • Priority for the application having the focus: 1 (Navigator)

• Applications executing:

Name	Supplier	Priority	Focus
Weather	Broadcaster X	2	No
Navigator	Decoder manufacturer	1	Yes
Shop	Broadcaster X	2	No

30

The advantages of the invention are numerous.

- A broadcaster or a supplier of services can himself define the behavior of a decoder, relating to the initiation of a downloaded application, through
35 management of the priorities of the instruction sets

and by including, for example, an instruction set in the preamble of the application, in such a way that this set can be executed while the application finishes being loaded.

5 - A broadcasting of management instruction sets by way of the service information of a digital stream makes it possible to define the conditions of initiation of applications, without the broadcasting of these sets necessarily having to be done at the same
10 time as that of the application.

 - The manufacturer of the hardware (decoder in the present case) can also monitor the behavior of an application. For example, by choosing the priorities appropriately, he can retain full control of the
15 decoder and force any application to use the predetermined management instruction set.

00007041 062004
F002000 F002000

1. Operating system
2. Execution system
- 5 3. Application
4. Applications manager
5. Digital decoder
6. Television
7. Satellite receiver
- 10 8. Cable network
9. Hertzian antenna
10. Virtual machine
11. Variable management instruction set
12. Loading means
- 15 13. User interface
14. Direct link with a source of applications
15. Application link
16. Application preamble